

This listing of claims will replace all prior versions, and listings of claims in the application:

**LISTING OF CLAIMS**

Claim 1 (Previously presented): A method for creating a dialog box visually differentiable from a displayed background, the method comprising:

receiving a command to create the dialog box, the command including a selected background color of the dialog box configured to have a value;

only using a reserved color associated with a reserved color value to draw a dialog box boundary surrounding the dialog box, the reserved color being a color reserved by an operating system of a platform to be used by the operating system only, the reserved color being indexed to a value of a single color table wherein the single color table includes an index for the selected background color and an index for a displayed background color; and

drawing a dialog box background using the value of the selected background color, wherein a drawn dialog box boundary is maintained in the reserved color associated with the reserved color value, and further wherein using the reserved color to draw the dialog box boundary is configured to constantly maintain visual differentiation between the dialog box and the displayed background.

Claim 2 (Previously presented): A method for creating a dialog box visually differentiable from a displayed background as recited in claim 1, wherein drawing the dialog box background using the selected background color value includes:

determining whether the value for the selected background color is equivalent to the reserved color value or a cross-platform compatible color value, the determining including,

mapping the value of the selected background color to a previously assigned cross-platform compatible color value when the selected background color value is

equivalent to the reserved color value; and

mapping the value of the selected background color to a corresponding cross-platform compatible color value when the selected background color value is equivalent to the cross-platform compatible color value.

**Claim 3 (Original):** A method for creating a dialog box visually differentiable from a displayed background as recited in claim 1, wherein the dialog box is displayed using a graphic image.

**Claim 4 (Previously presented):** A method for creating a dialog box visually differentiable from a displayed background as recited in claim 1, wherein the dialog box boundary is configured to include a slider, a border, text, a button, or a scroll bar.

**Claim 5 (Original):** A method for creating a dialog box visually differentiable from a displayed background as recited in claim 4, wherein the dialog box boundary is a border.

**Claim 6 (Original):** A method for creating a dialog box visually differentiable from a displayed background as recited in claim 5, wherein the border is beveled.

**Claim 7 (Original):** A method for creating a dialog box visually differentiable a displayed background on a display system as recited in claim 1, wherein the dialog box is a Java-based dialog box.

**Claim 8 (Previously presented):** A method for selecting colors to draw a dialog

box having a visually differentiable boundary, the method comprising:

determining whether a dialog box boundary, a dialog box background, or a dialog box component is being drawn, the determining including,

only selecting a reserved color when drawing the dialog box boundary surrounding the dialog box by bypassing a mapping of the reserved color to a previously assigned cross-platform compatible color;

only selecting a cross-platform compatible color when drawing the dialog box background, wherein an index of the reserved color, an index for the selected background color and an index for a displayed background color are stored within a single color table; and

selecting a cross-platform compatible color when drawing the component contained within the dialog box,

wherein the drawn dialog box boundary is maintained in the selected reserved color so long as the dialog box is displayed, and further wherein the bypassing the mapping of the reserved color to a previously assigned cross-platform compatible color is configured to draw a dialog box having a differentiable boundary.

Claim 9 (Previously presented): A method for selecting colors to draw a dialog box having a visually differentiable boundary as recited in claim 8, wherein the reserved color is a color reserved by an operating system of a platform to only be used by the operating system.

Claim 10 (Previously presented): A method for selecting colors to draw a dialog box having a visually differentiable boundary as recited in claim 8, wherein only selecting the cross-platform compatible color when drawing the dialog box background includes:

using a value of the selected background color to map the selected background to a previously assigned cross-platform compatible color when the value of the selected background color is equivalent to a reserved color value; and

using the value of the selected background color to map the selected background color value to a corresponding cross-platform compatible color when the value of the selected background color is equivalent to a cross-platform compatible color value.

Claim 11 (Previously presented): A method for selecting colors to draw a dialog box having a visually differentiable boundary as recited in claim 8, wherein the dialog box is a JAVA-based dialog box, a C-based dialog box, or a C++-based dialog box.

Claim 12 (Original): A method for selecting colors to draw a dialog box having a visually differentiable boundary as recited in claim 8, wherein the dialog box is displayed using a graphic image.

Claim 13 (Original): A method for selecting colors to draw a dialog box having a visually differentiable boundary as recited in claim 8, wherein the colors selected to draw the dialog box boundary, dialog box background, and components contained within the dialog box are processed by a controller.

Claim 14 (Original): A method for selecting colors to draw a dialog box having a visually differentiable boundary as recited in claim 13, wherein the controller is integrated in a graphics card.

Claim 15 (Previously presented): A method for generating dialog box graphical user

interfaces (GUIs) that are presented over an underlying background image, comprising:

receiving a command to generate a dialog box;

if a boundary element of the dialog box is to be generated, the method includes, only implementing a reserved color for the generation, the reserved color not being available for use in generating graphical context of background color of the dialog box, wherein an index of the reserved color, an index for the background color and an index for a display background color are stored within a single color table, and

wherein the generated boundary element is maintained in the reserved color so long as the generated dialog box is displayed over the underlying background image.

Claim 16 (Previously presented): A method for generating dialog box graphical user interfaces (GUIs) that are presented over an underlying background image as recited in claim 15, further comprising:

if a background element of the dialog box is to be generated, the method includes, only implementing a cross-platform compatible color for the generation.

Claim 17 (Previously presented): A method for generating dialog box graphical user interfaces (GUIs) that are presented over an underlying background image as recited in claim 15, wherein the boundary element is configured to include a slider, a border, text, a button, or a scroll bar.

Claim 18 (Original): A method for generating dialog box graphical user interfaces (GUIs) that are presented over an underlying background image as recited in claim 17, wherein the boundary element is a border.

**Claim 19 (Original):** A method for generating dialog box graphical user interfaces (GUIs) that are presented over an underlying background image as recited in claim 18, wherein the border is beveled.

**Claim 20 (Previously presented):** A method for generating dialog box graphical user interfaces (GUIs) that are presented over an underlying background image as recited in claim 15, wherein the dialog box is a JAVA-based dialog box, a C-based dialog box, or a C++-based dialog box.